In the Claims

Claims 2, 5 and 12-33 are cancelled without prejudice.

Claim 1 is amended.

Claims 1, 3, 4 and 6-11 are pending and are listed below:

1. (Currently Amended) A method comprising:

loading one or more source processing chains to support execution of a development project, the source processing chains comprising a series of filters to process and render media content, wherein loading of the one or more processing chains comprises:

identifying which source(s) will be required to support execution of the next M seconds of the development project, wherein M is at least as long as necessary to construct a processing chain;

searching one or more cache(s) to determine whether the source processing chain(s) associated with the source(s) are available within the one or more cache(s); and

retrieving the one or more processing chains from a memory location denoted by an associated one or more pointers in the cache for integration with the development project; and

determining whether each of the one or more processing chains will be subsequently required during execution of this or another development project and, if so, caching those filter chains which will be subsequently required.

2. (Canceled).

3. (Original) A method according to claim 2, further comprising:

determining whether processing chain(s) retrieved from the cache(s) satisfy processing requirements of the development project; and

modifying one or more objects of one or more of the processing chain(s) retrieved from the cache(s) that do not satisfy the processing requirements of the development project.

4. (Original) A method according to claim 3, wherein modifying one or more objects may comprise one or more of adding processing objects to the processing chain(s), removing one or more processing objects from the processing chain(s), or changing one or more operating attributes of one or more processing objects within the processing chain(s).

5. (Canceled).

6. (Original) A method according to claim 1, wherein determining whether a processing chain will subsequently be required comprises:

determining whether any future calls to a source coupled to the processing chain exist within this development project; and

determining whether any future calls to a source coupled to the processing chain may be received during execution of future development projects.

7. (Original) A method according to claim 6, wherein it is assumed that each processing chain may well be required to support future execution of this or a future development project.

8. (Original) A method according to claim 1, wherein caching the processing chain comprises:

assigning the processing chain a unique identifier; and

storing the unique identifier along with a pointer to a memory location occupied by the processing chain in a cache.

9. (Original) A method according to claim 8, wherein the unique identifier is one or more of a source file handle, a source file name, a random numeric identifier uniquely assigned to the processing chain, a graphical icon, an alphanumeric character, and the like.

10. (Original) A storage medium comprising a plurality of executable instructions which, when executed, implement a method according to claim 1.

11. (Original) A computing system comprising:

a storage medium having stored therein a plurality of executable instructions; and

an execution unit, coupled to the storage medium, to execute at least a subset of the plurality of executable instructions to implement a method according to claim 1.

12.-33. (Canceled).